



10555/023001

1 / 27

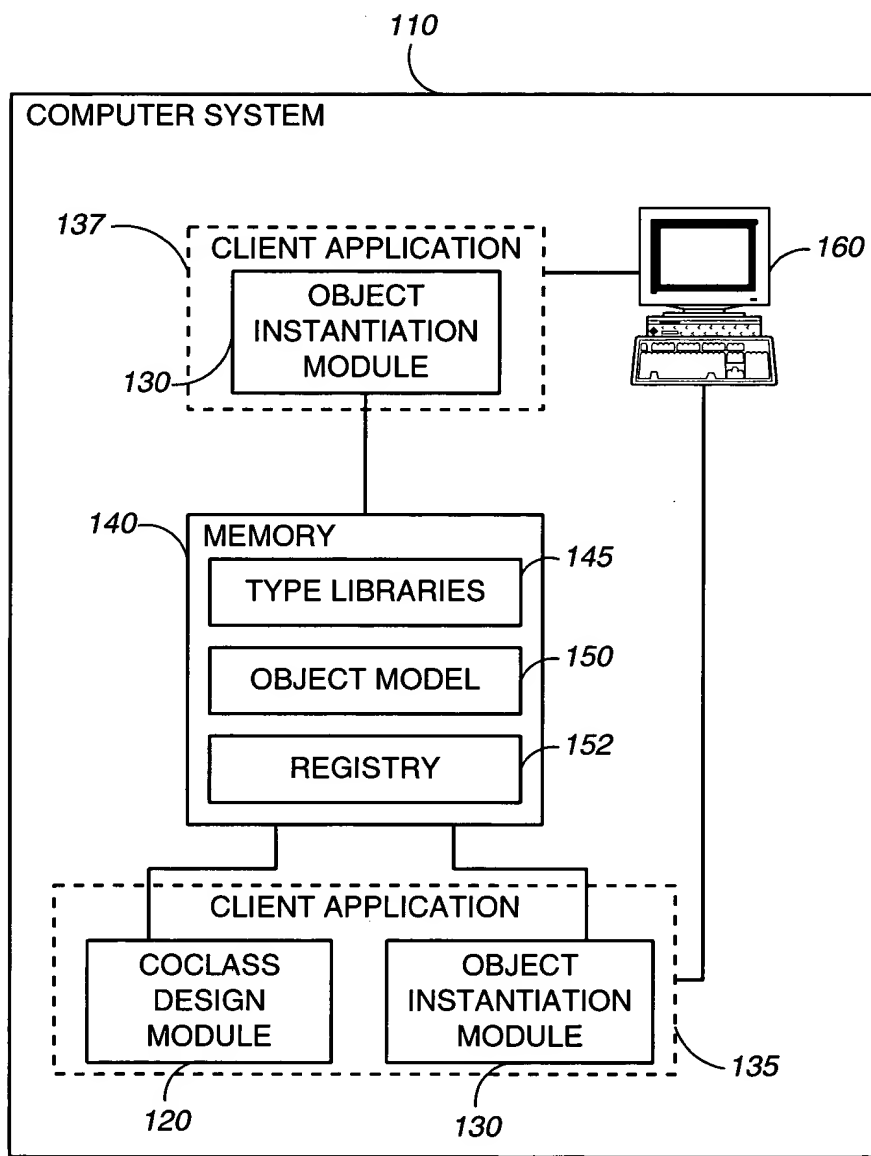
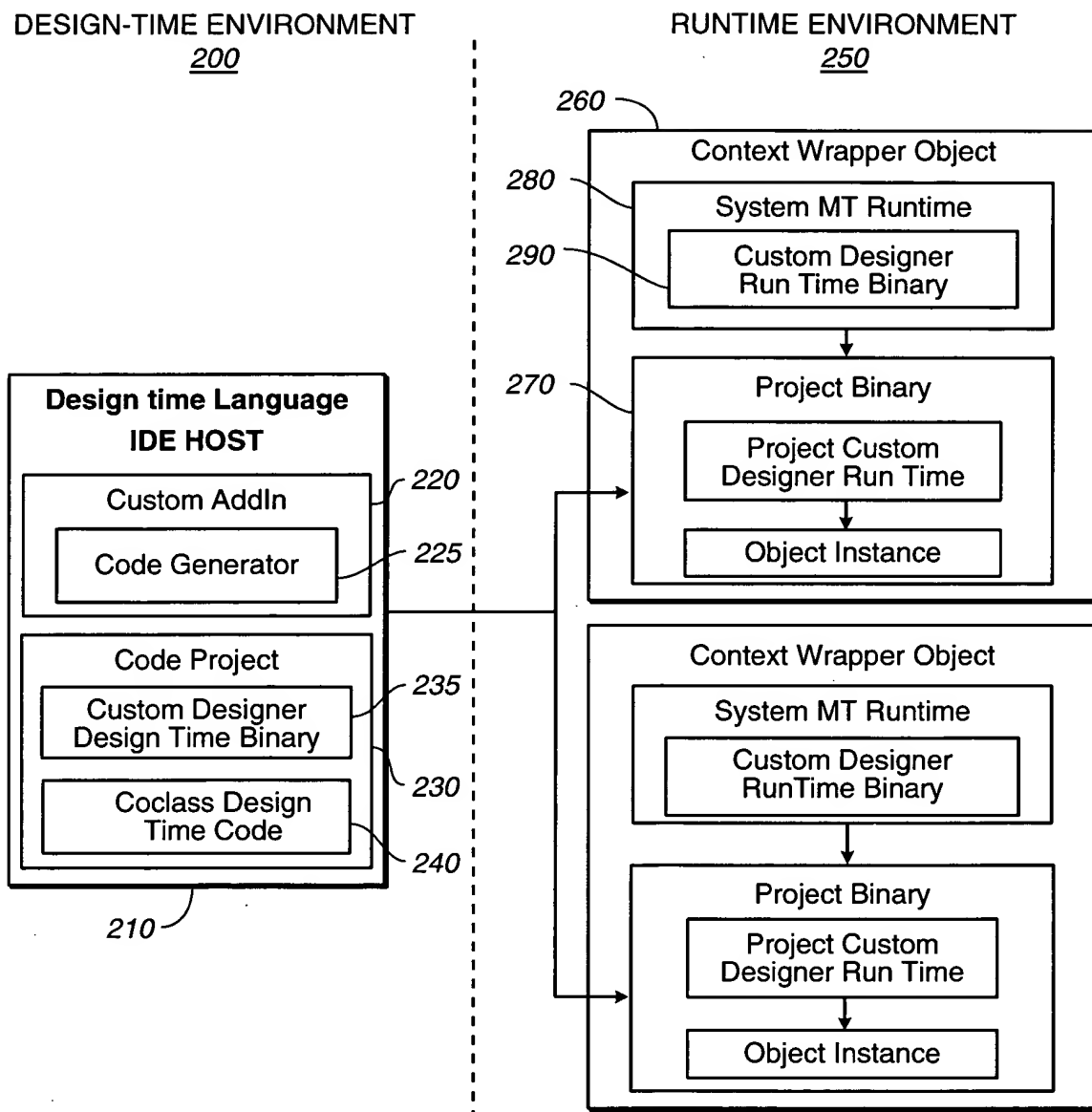
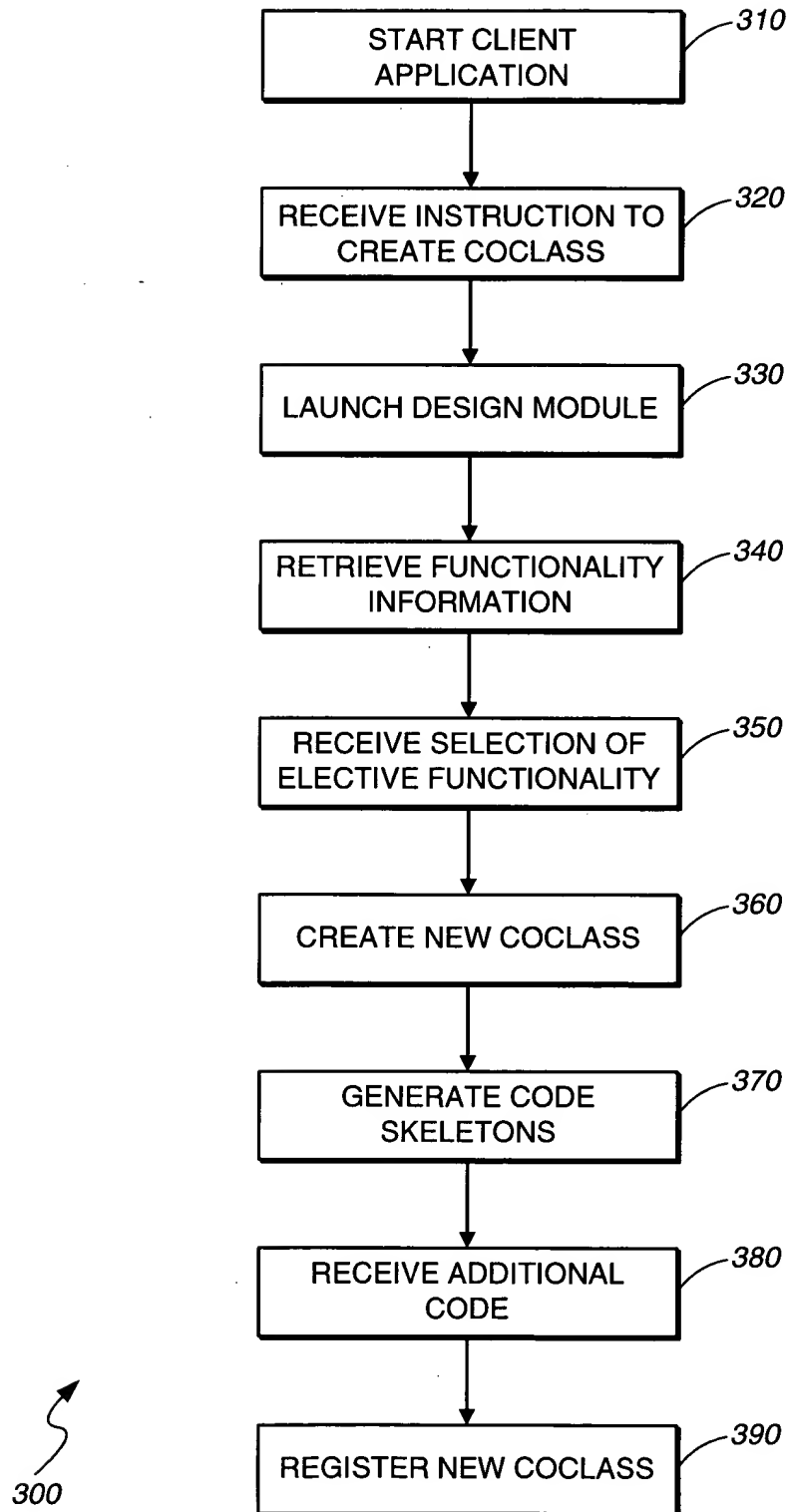


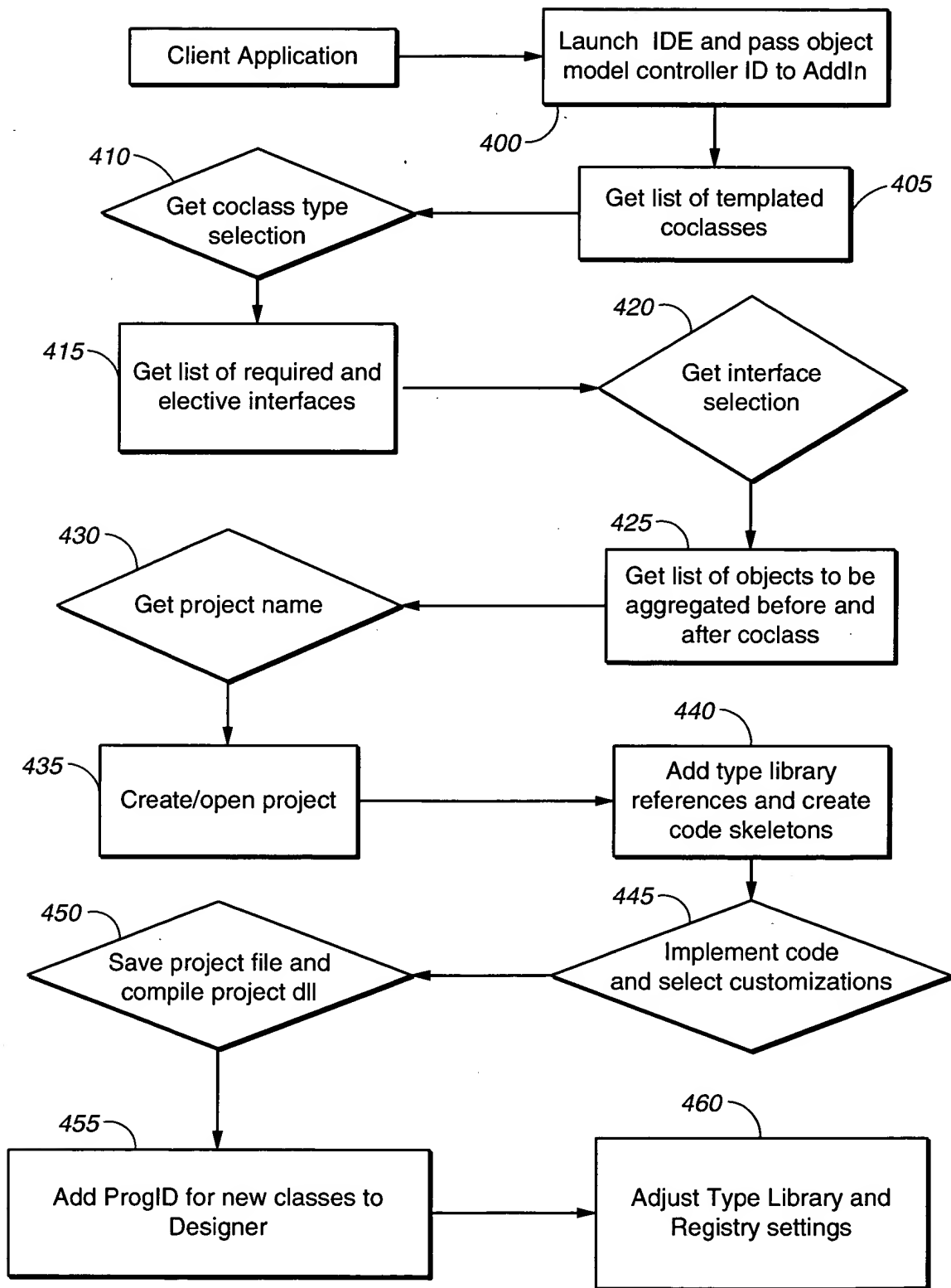
FIG._1

100

**FIG._2**

3 / 27

**FIG. 3**

**FIG. 4A**

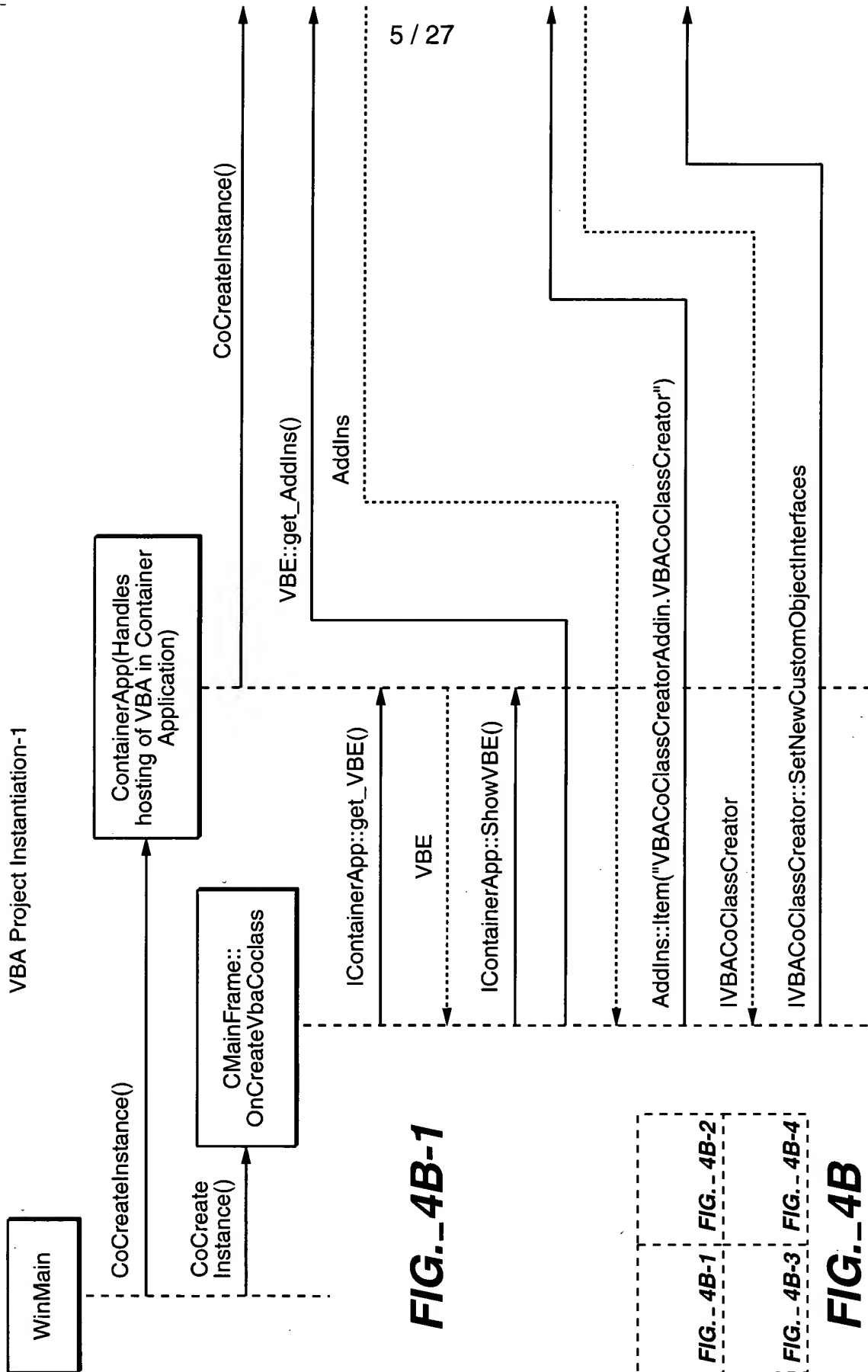


FIG._4B-1

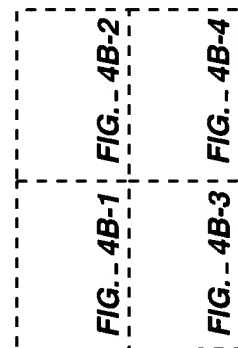
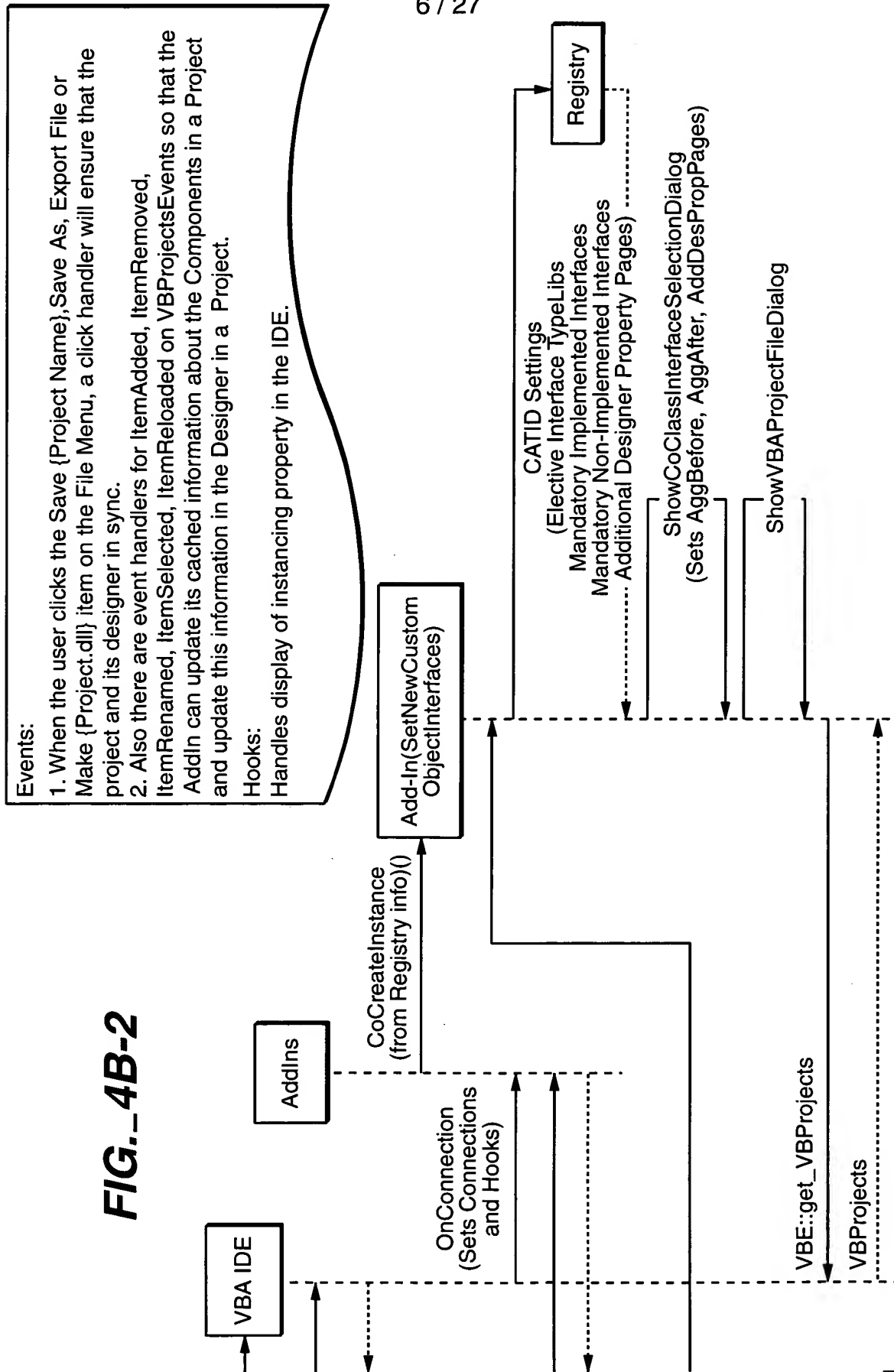


FIG._4B

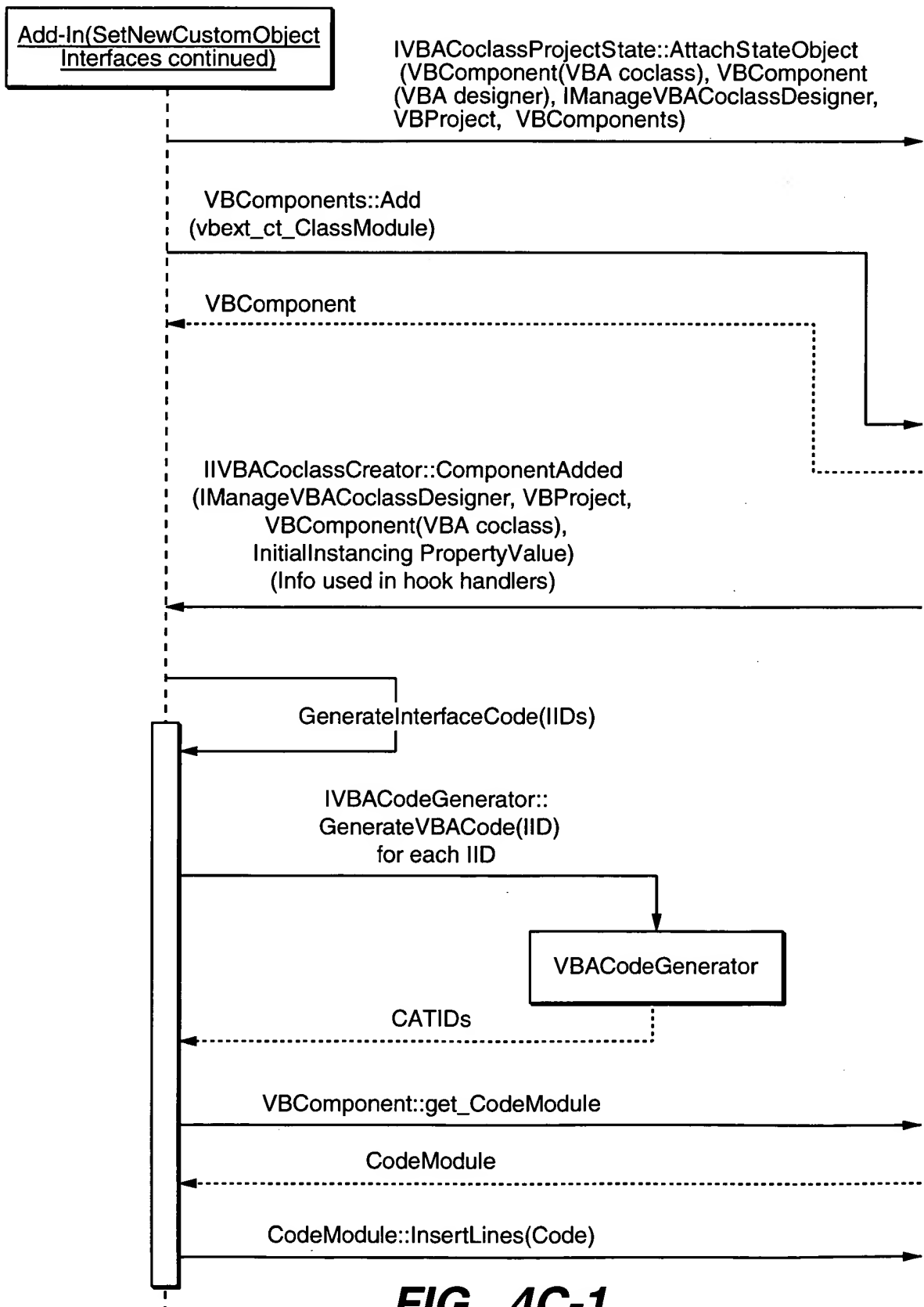




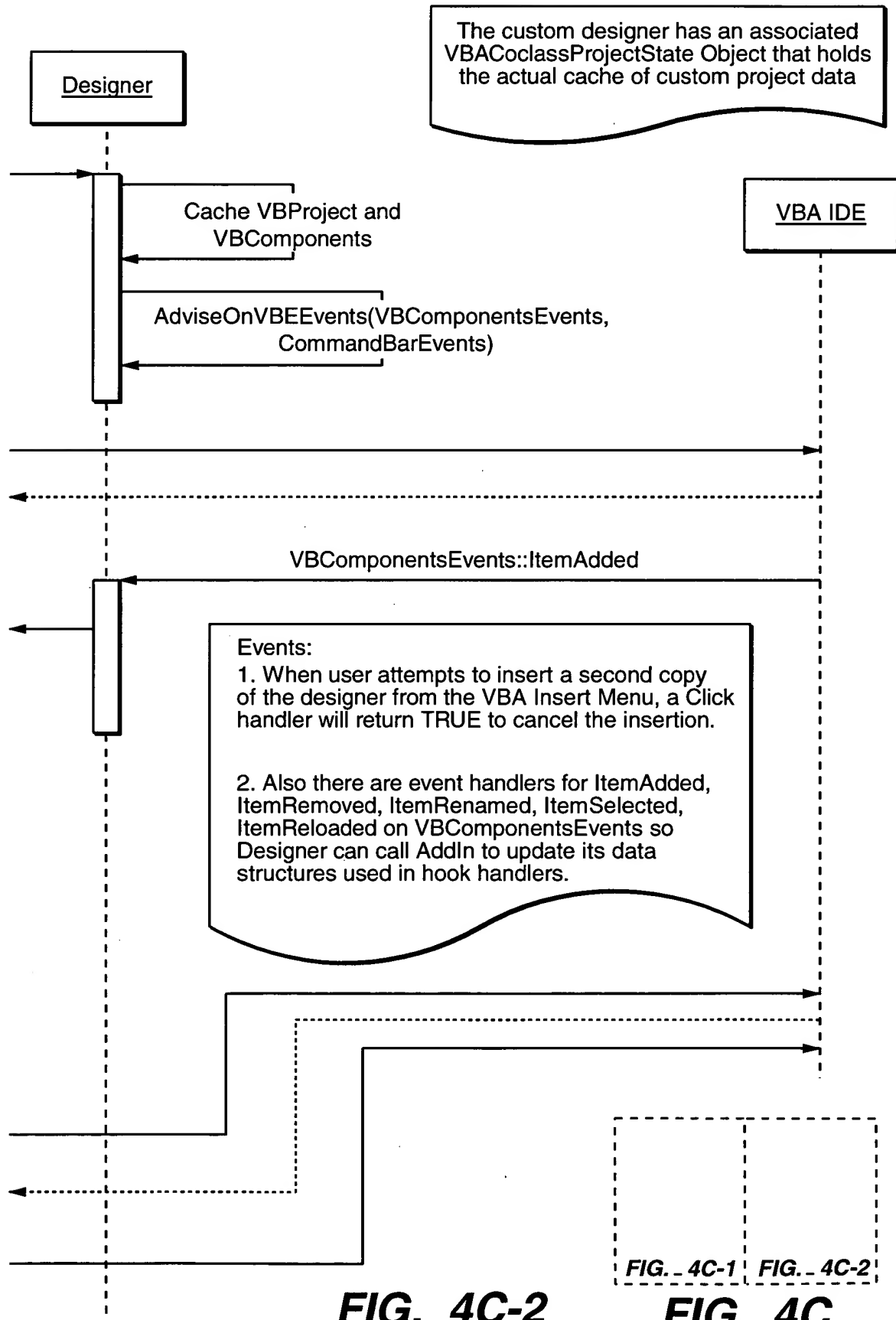
To allow alternate storage such as to a database for now it will be necessary to persist the project file to a disk file first because the MT implementation only allows persistence in a disk file.

FIG._4B-3

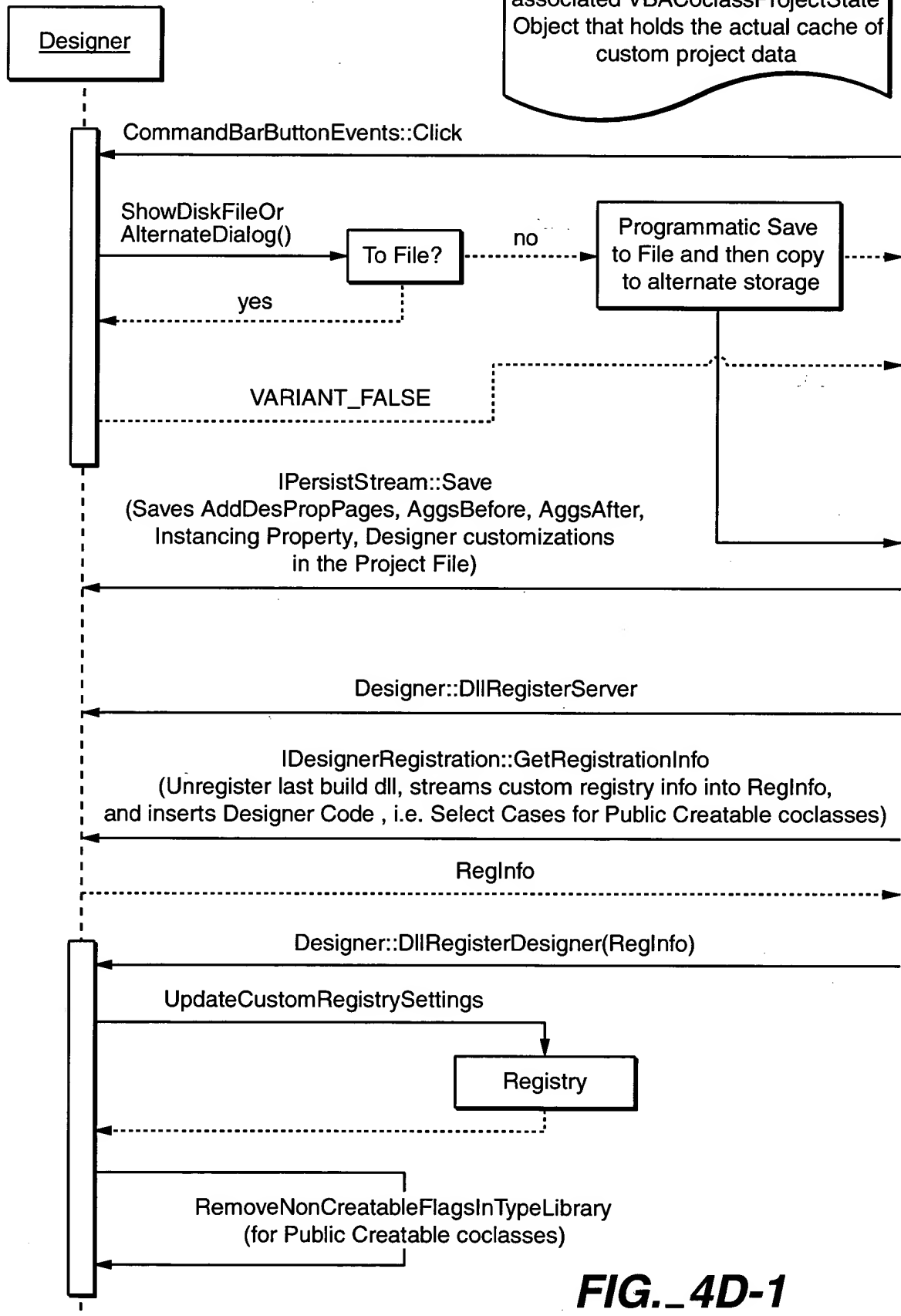


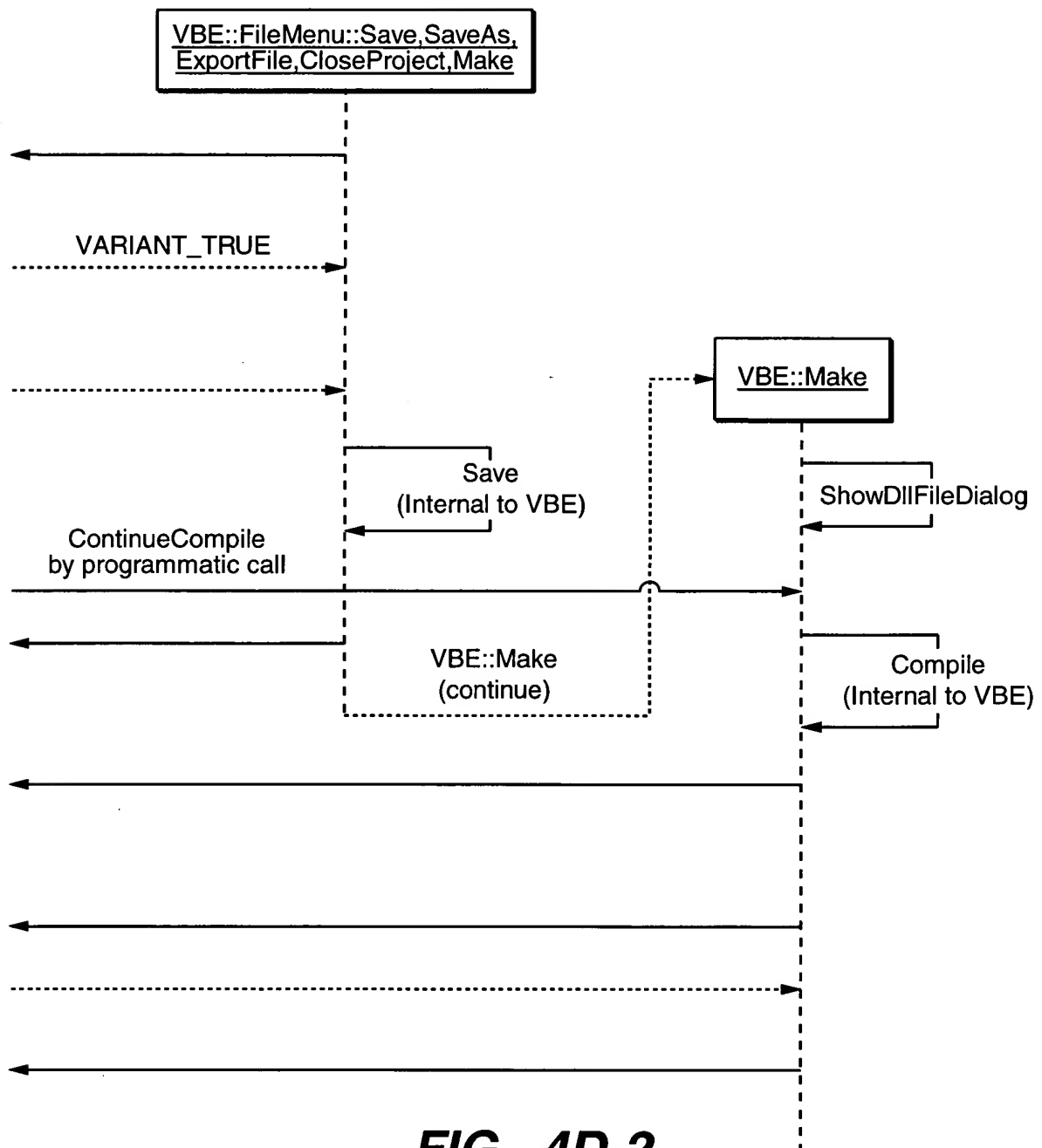
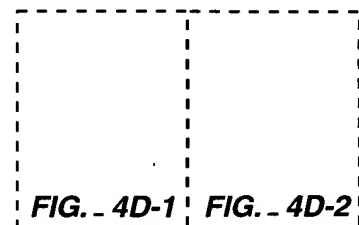
**FIG. 4C-1**

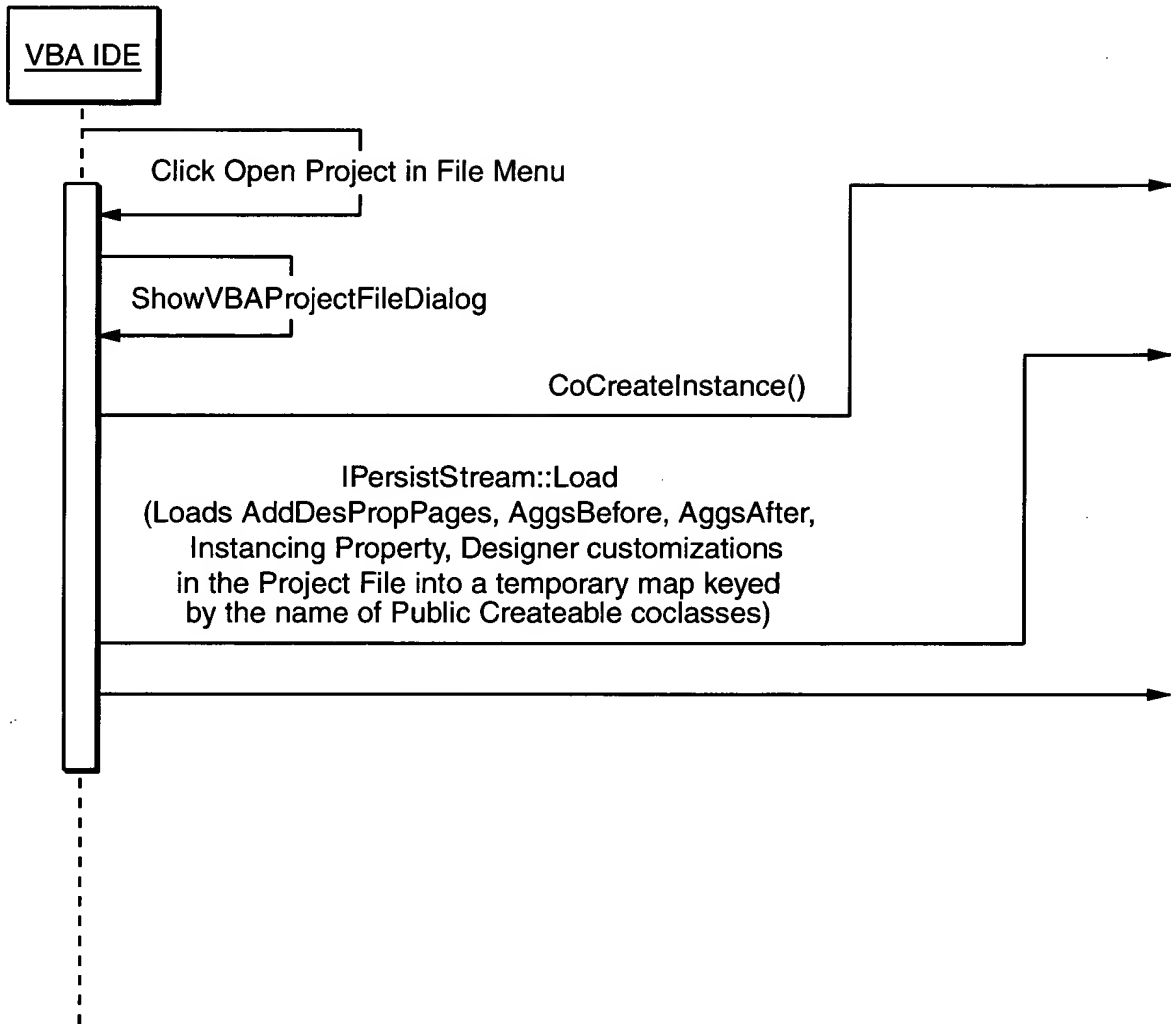
10 / 27



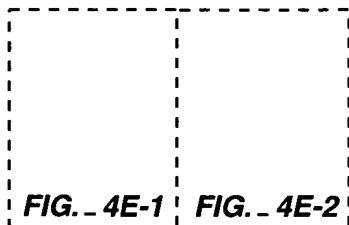
The custom designer has an associated VBACoclassProjectState Object that holds the actual cache of custom project data

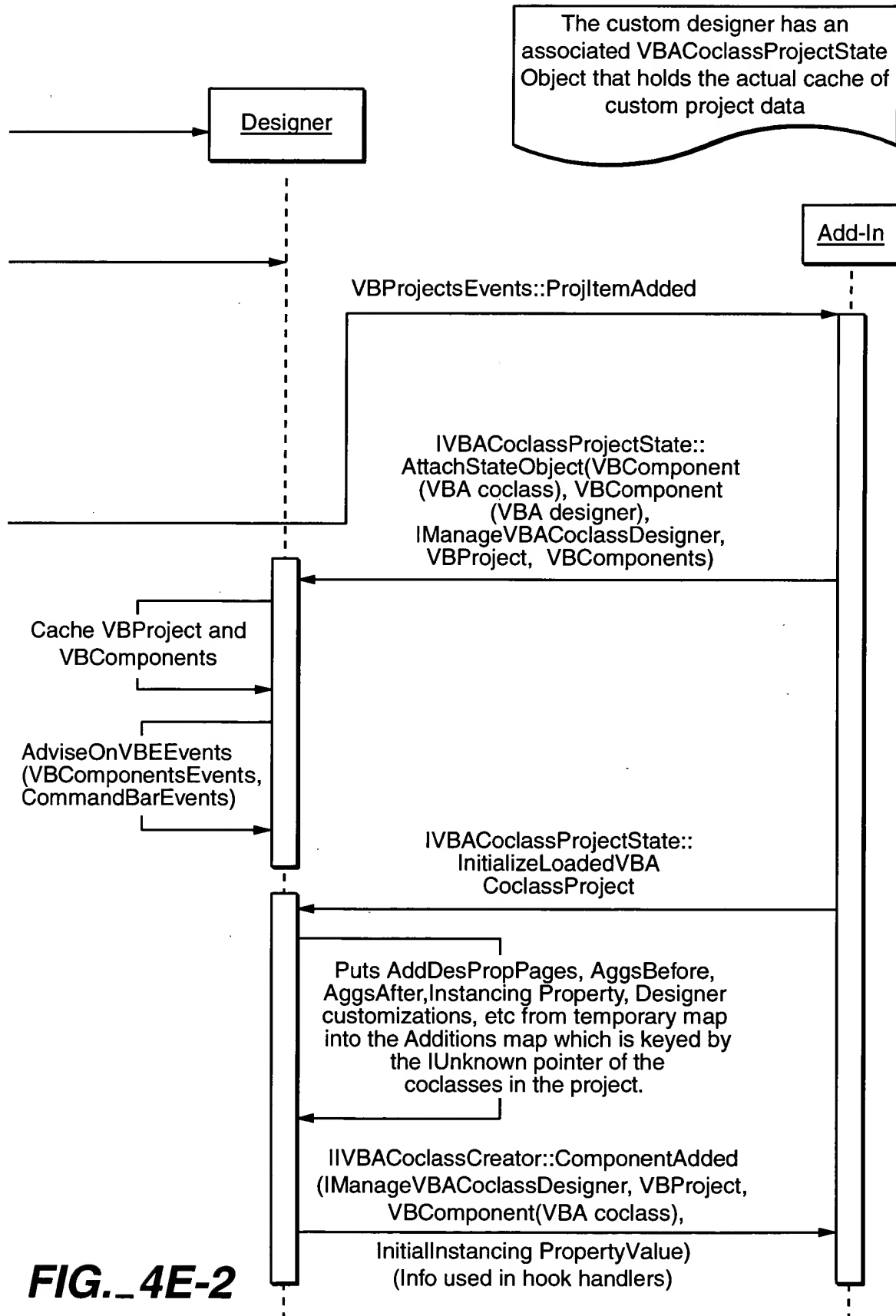
**FIG._4D-1**

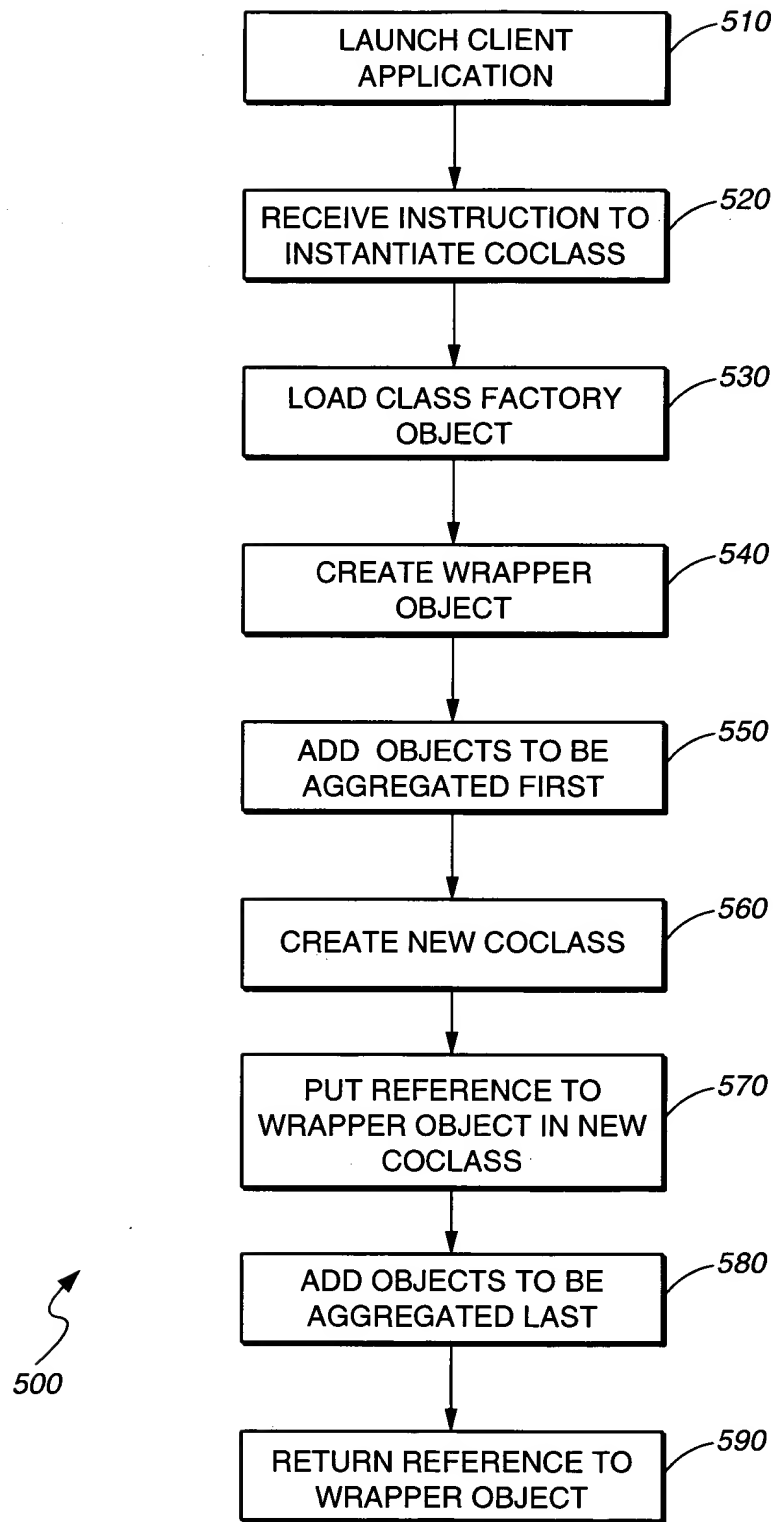
**FIG._4D-2****FIG._4D**



When the user selects deleting a project from the VBA IDE, the VBProjectsEvents::ProjItemRemoved handler in the AddIn removes the project's custom settings in the data structures in the AddIn. Also when the user inserts/deletes a designer into/from a project, or inserts/deletes a class into/from a project, the VBCommandBarEvents::Click event handler in the AddIn and designer, respectively, add to or delete from the data caches in the AddIn and designer.

FIG._4E-1**FIG._4E**

**FIG._4E-2**

**FIG. 5**

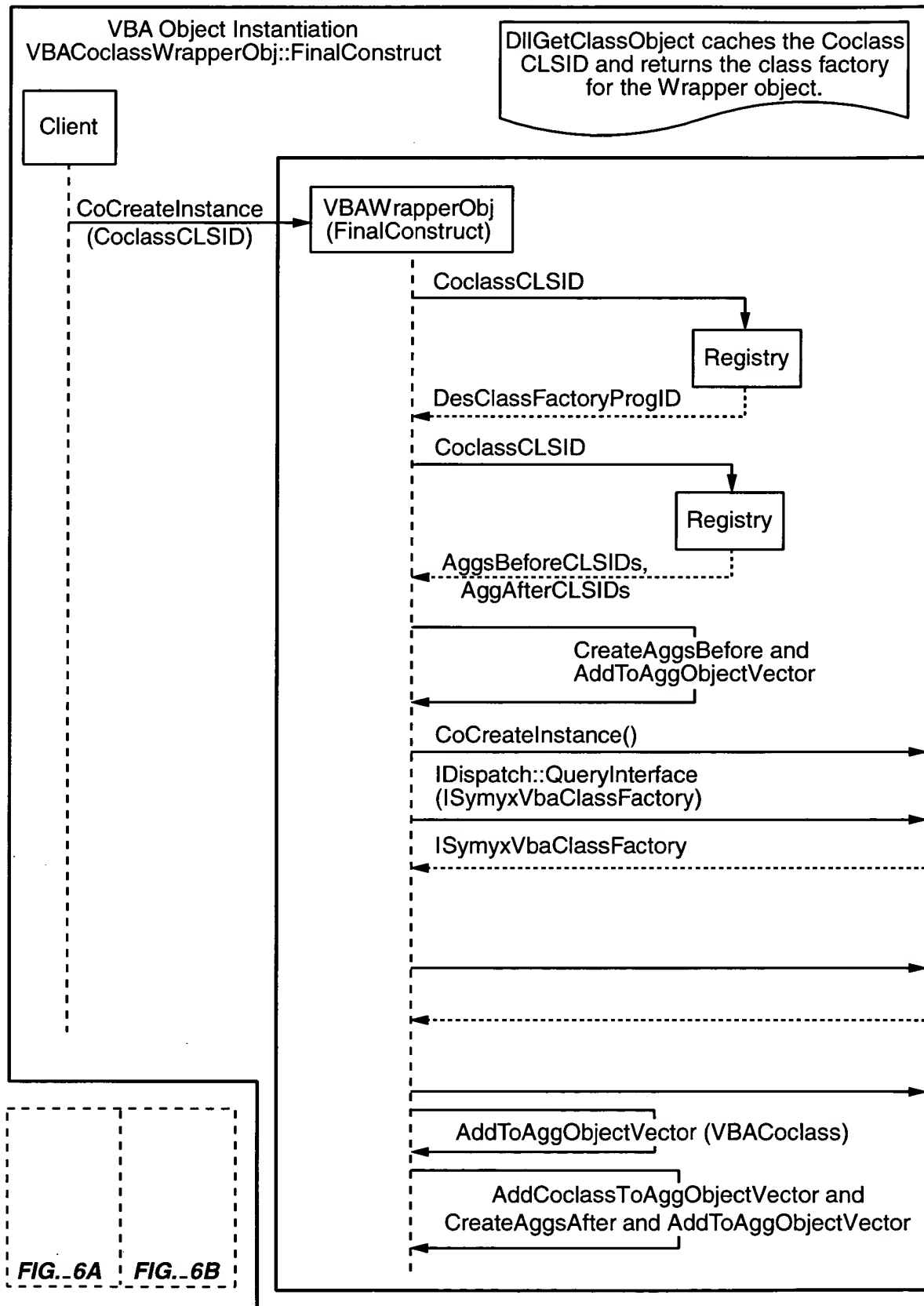
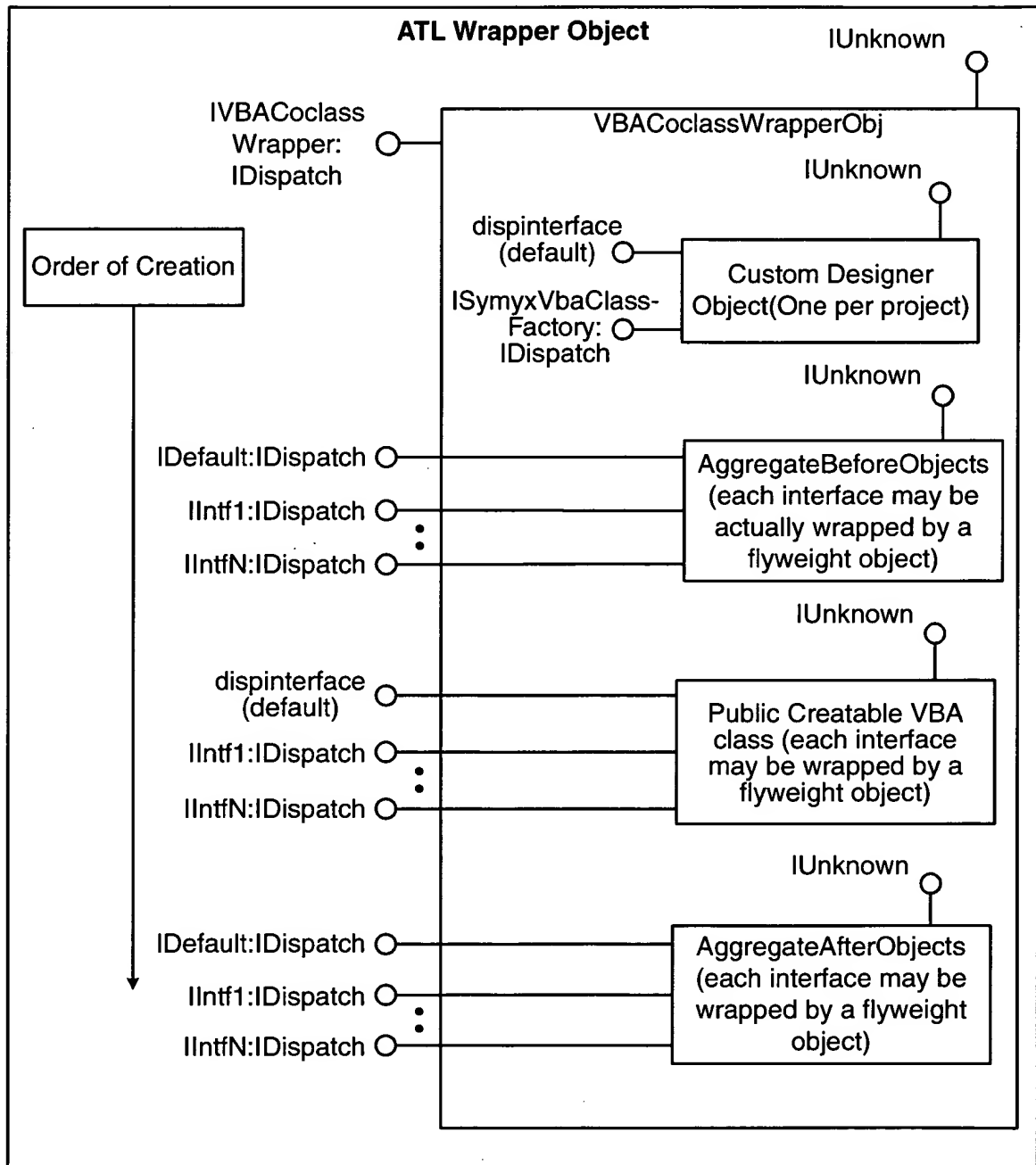
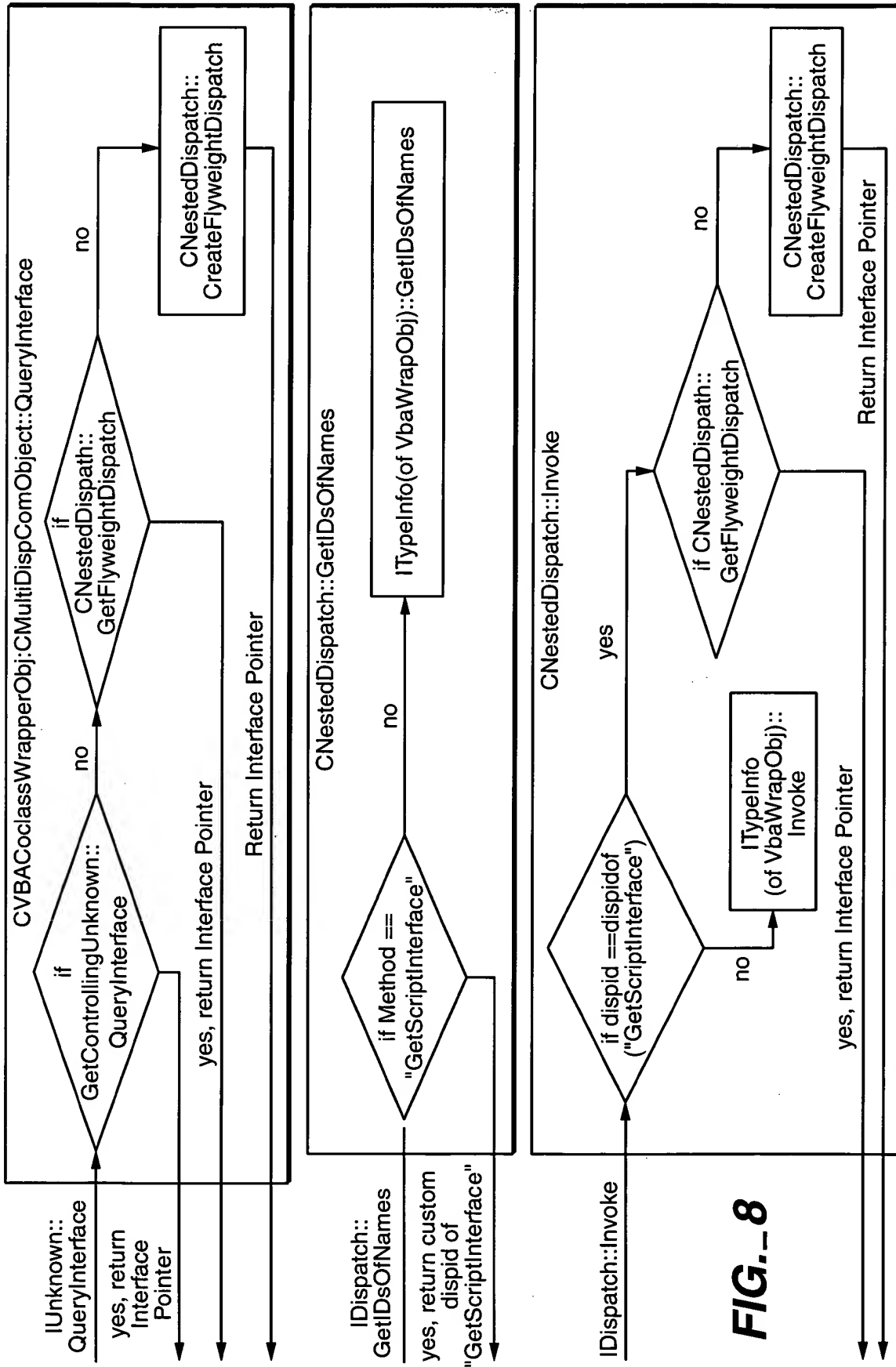
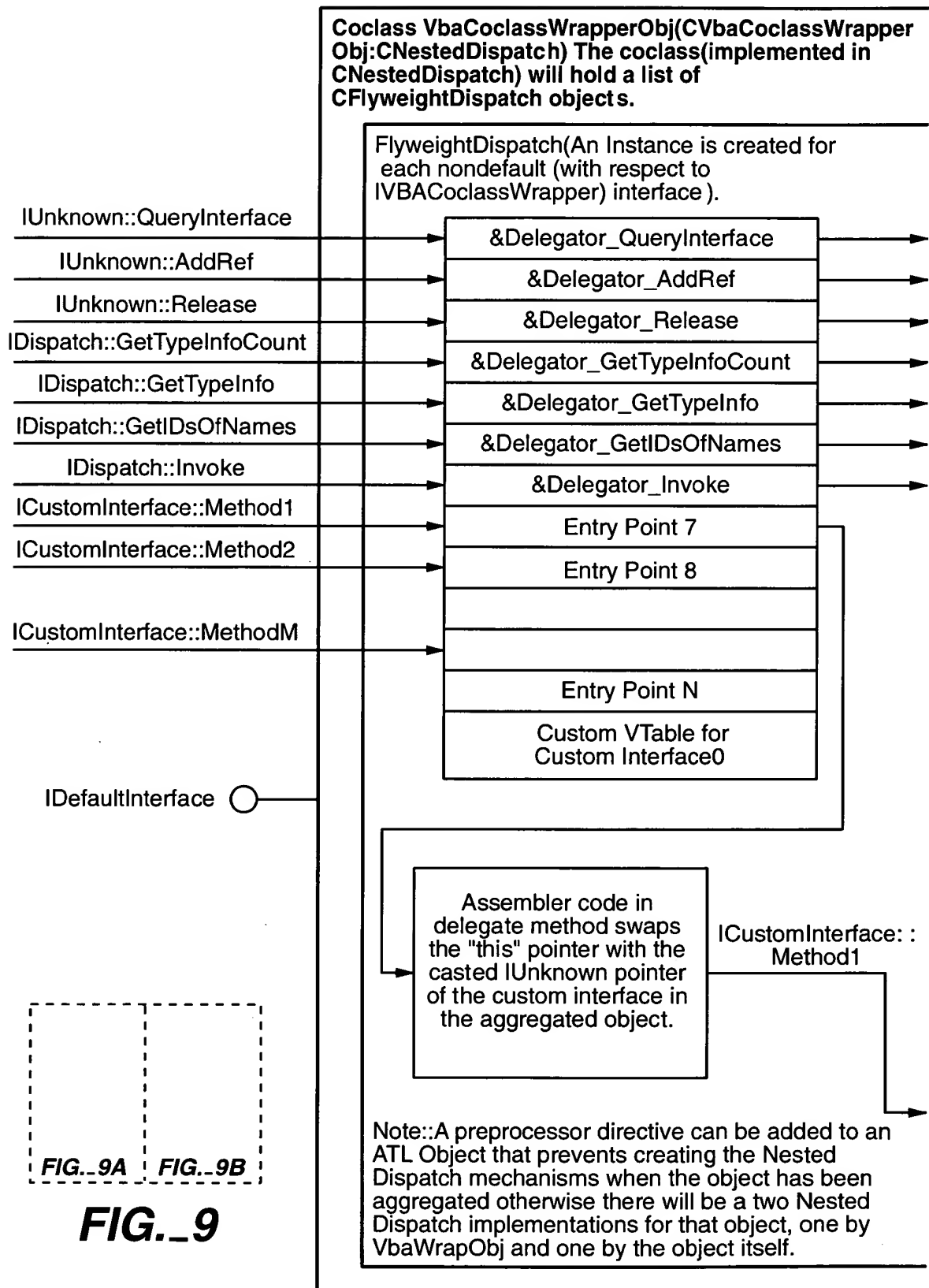


FIG._6

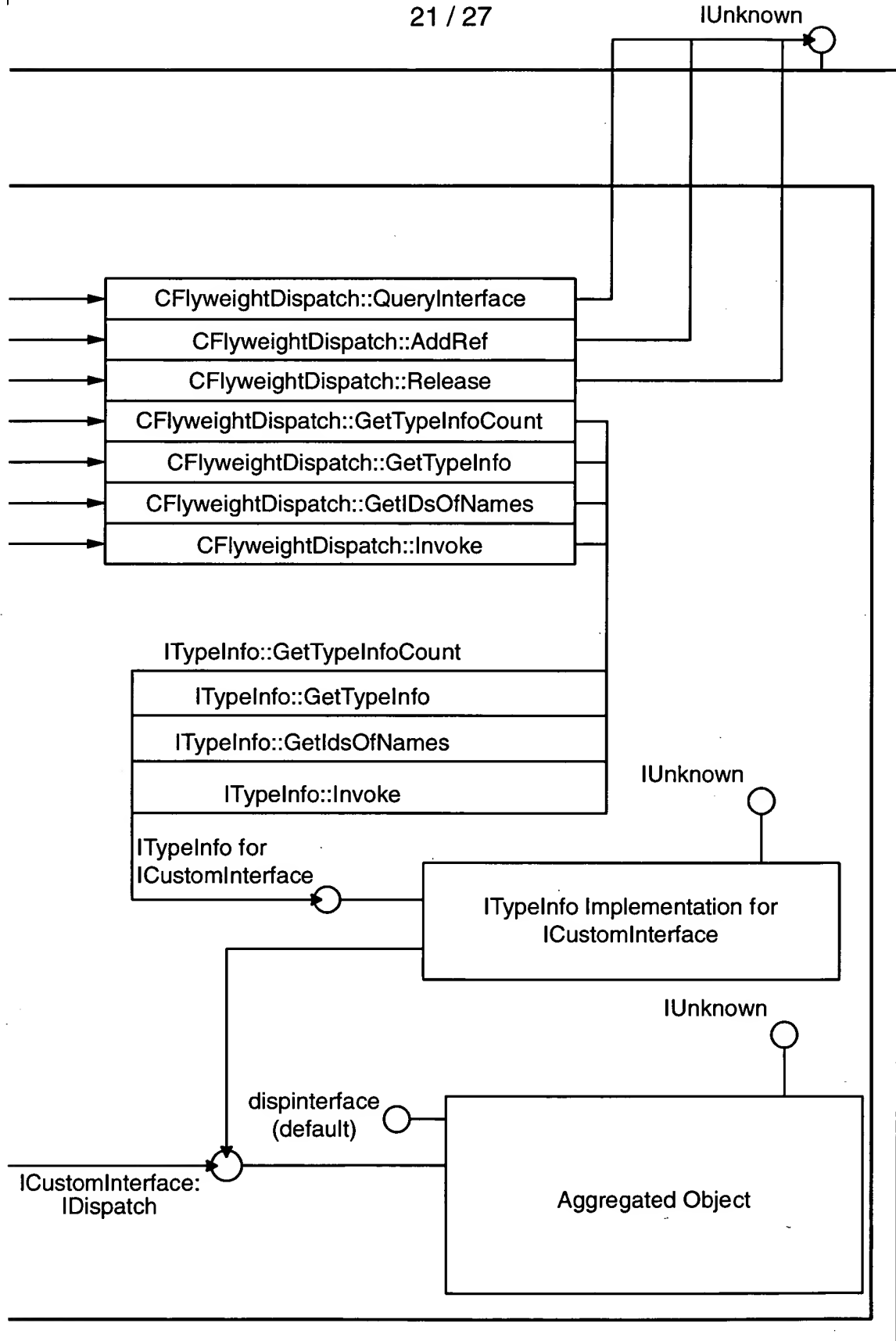
FIG._6A

**FIG. 7**

**FIG._8**



21 / 27

**FIG. 9B**

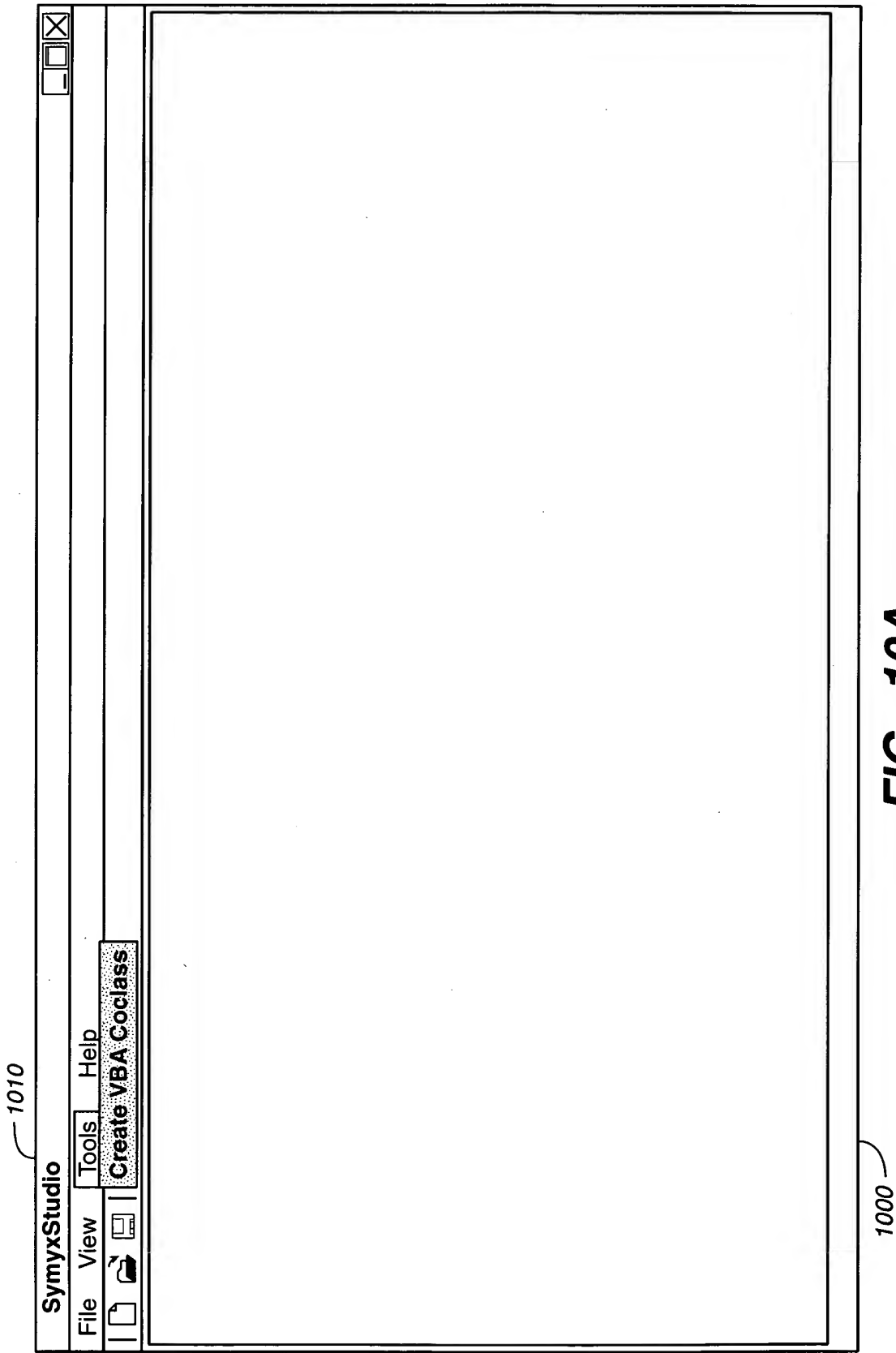


FIG. 10A

Interface Selection Dialog

Impressionist Object Model Controller

Symyx Impressionist Custom Resource

Interface	Type Library	Status
<input checked="" type="checkbox"/> INode	iiCore	Mandatory Implemented
<input type="checkbox"/> ISupportDynamicProperties	iiCore	Elective
<input type="checkbox"/> IHPLCPump	iiHPLCPump	Elective
<input type="checkbox"/> IPump	iiPump	Elective
<input type="checkbox"/> IDigitalInput	iiDigital	Elective
<input checked="" type="checkbox"/> IValve	iiValve	Elective
<input type="checkbox"/> IPressureInput	iiPressure	Elective

OK

Cancel

FIG._ 10B

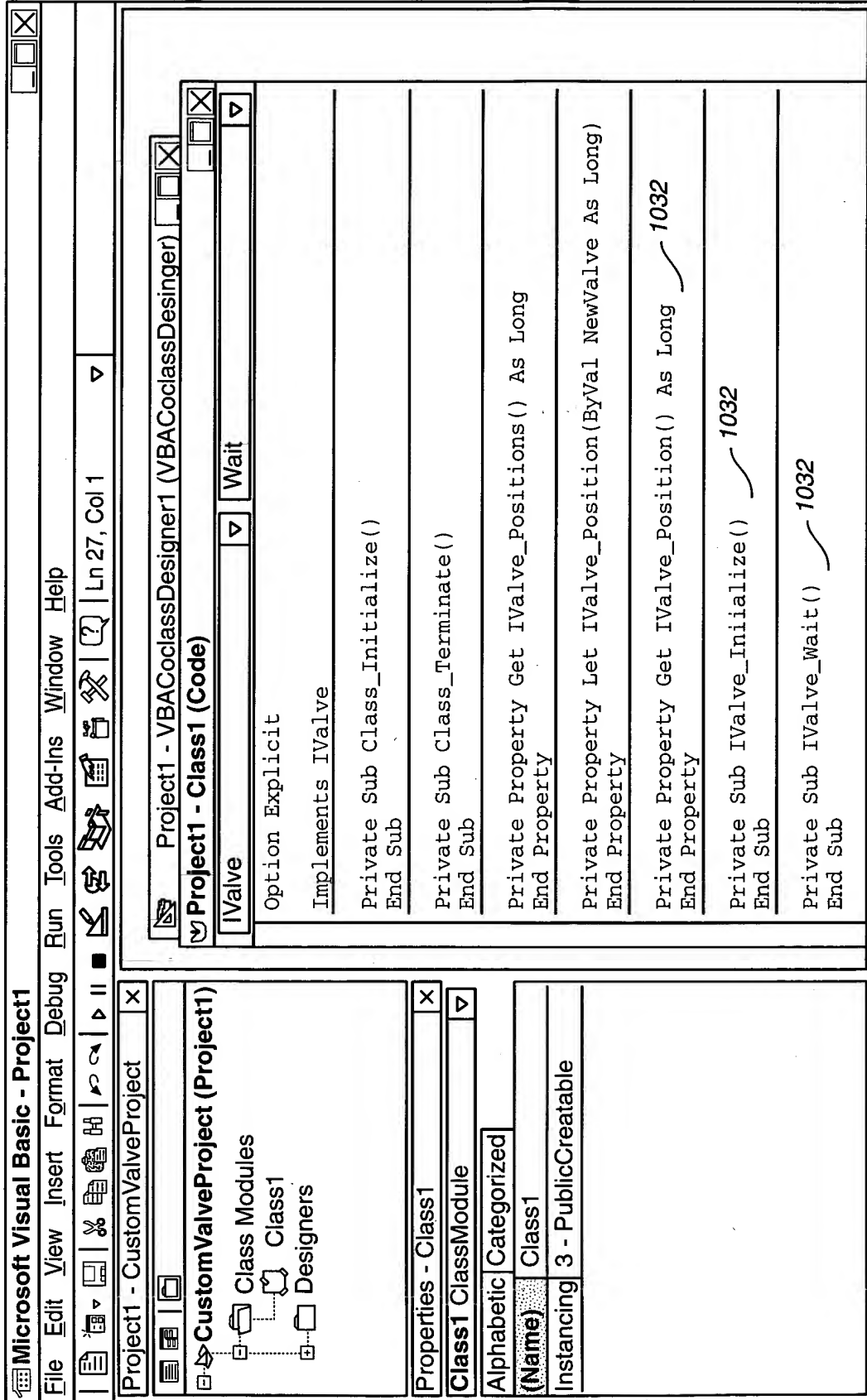


FIG.. 10C

Microsoft Visual Basic - F:\VBIntegrationDemo\Custom\ValveProject.vba

File Edit View Insert Format Debug Run Tools Add-Ins Window Help

Project1 - CustomValveProject

CustomValveProject (F:\VBIntegration Demo\Cust

Modules

TheContextObjectModule

Class Modules

Class1

Designers

Properties - Class1

Class1 ClassModule

Alphabetic Categorized

(Name) Class1

Instancing 3 - PublicCreatable

F:\VBIntegrationDemo\CustomValveProject.vba - Class1 (Code)

Class

Initialize

Implements IValve

Private m_ValvePosition As Long

Private m_Address As IPropertyInteger

Private m_NumPositions As IPropertySingleChoice

Private Sub Class_Initialize()

Dim AggNode As INode

Set AggNode = TheContextObject

AggNode.Version = 1

AggNode.IconResID = 0

AggNode.Name = "Dave's Custom Valve"

Set m_Address = New PropertyInteger

m_Address.Name = "Address"

m_Address.XmlTag = "Address"

m_Address.Valve = 1

AggNode.Properties.Add m_Address

Set m_NumPositions = New PropertyEnum

m_NumPositions.Name = "Number of Positions"

m_NumPositions.XmlTag = "NumPositions"

m_NumPositions.Choices = "2|4|8|16"

m_NumPositions.Text = "2"

AggNode.Properties.Add m_NumPositions

m_ValvePosition = 1

End Sub

Private Property Get IValve_Positions() As Long

IValve_Positions = CInt(m_NumPositions.Text)

End Property

Private Property Let IValve_Position(ByVal newValue As Long)

m_ValvePosition = newValue

End Property

Private Property Get IValve_Position() As Long

IValve_Position = m_ValvePosition

End Property

1040

FIG.-- 10D

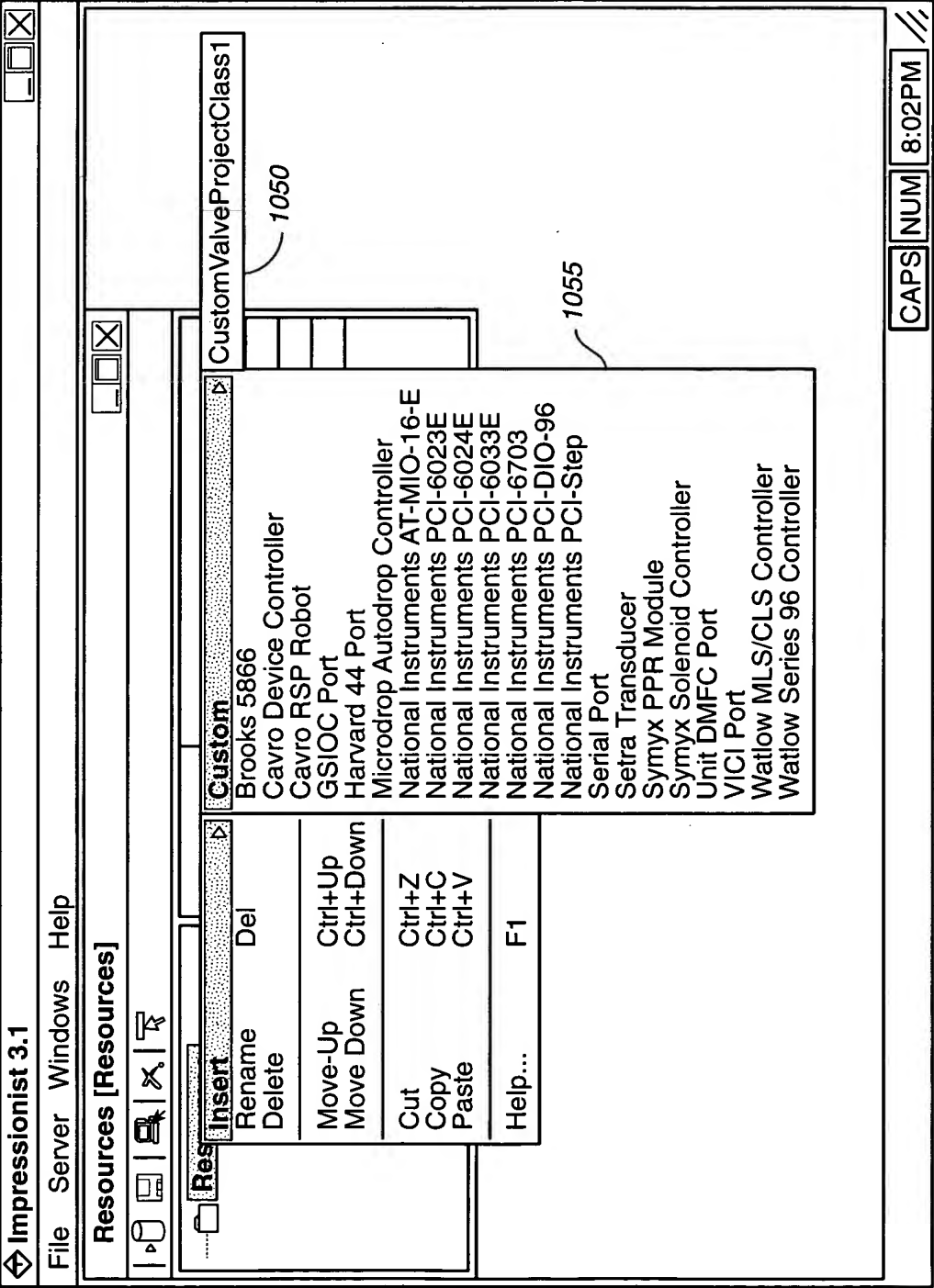


FIG. 10E